[redacted]

UNITED STATES DISTRICT COURT
FOR THE EASTERN DISTRICT OF TEXAS
MARSHALL DIVISION

SEVEN NETWORKS, LLC, Plaintiff, v. GOOGLE LLC, Defendant.	CIVIL ACTION NO. 2:17-CV-442-JRG LEAD CASE PATENT CASE JURY TRIAL DEMANDED
v. SAMSUNG ELECTRONICS AMERICA, INC. AND SAMSUNG ELECTRONICS CO., LTD., Defendants.	CIVIL ACTION NO. 2:17-CV-441-JRG CONSOLIDATED CASE

SEVEN'S MOTION TO COMPEL GOOGLE TO PRODUCE SOURCE CODE

[redacted]

INTRODUCTION

Understandably, Google is protective of its source code. To address that legitimate concern, the Court entered a Protective Order that includes seven pages of heightened source code protections. *See* ECF No. 88 ¶ 10. But for Google, that is not enough. So its general strategy is to "produce incomplete bits and pieces of code and documentation" until compelled to do otherwise. *See PersonalWeb Techs., LLC v. Google, Inc.*, No. 6:11-CV-656, 2013 WL 12156505, at *1 (E.D. Tex. June 28, 2013) (quotation marks omitted) (compelling Google to produce source code). Google has undertaken considerable efforts to select and produce only portions of source code that Google deems relevant, as opposed to producing the entire code for the accused products. Google's obstructionist strategy "offends the discovery principles driving this District's patent rules." *See Edward D. Ioli Trust v. Avigilon Corp.*, No. 2:10–cv–605–JRG, 2012 WL 5830711, at *3 (E.D. Tex. Nov. 16, 2012).

In view of "the broad nature of discovery under the Federal Rules, Google's current level of production is insufficient." *Id.* at *2 n.2. Accordingly, SEVEN asks the Court to compel a complete production of the following source code for all versions of the accused devices and instrumentalities for the relevant damages period:

- device-side code for accused Nexus and Pixel products;
- application code for Google Drive, Google Maps, and Google+; and
- server code for Gmail, the Google Play Store, and the Google Play Console.

BACKGROUND

SEVEN has accused Google mobile devices (i.e., phones and tablets, which include Nexus and Pixel products) with Lollipop, Marshmallow, Nougat, or Oreo versions of the Android operating systems; the Google Play Store, along with its hardware, software, and

[redacted]

associated transactions; and all versions of Google's 2-Step Verification system, of infringing SEVEN's patents. The patents at issue are directed to conserving battery power on mobile devices, customizing mobile applications to the specifications of a user's device, providing network services to mobile devices, and user authentication.

The Patent Rules required Google to produce "[s]ource code . . . *sufficient to show* the operation of *any aspects or elements* of [the] Accused Instrumentalit[ies]" with its November 15, 2017 Invalidity Contentions. *See* P.R. 3-4(a) (emphases added). But rather than provide complete source code with its Invalidity Contentions, Google instead produced code for only specific functionalities. Since that time, the SEVEN has attempted unsuccessfully for months to secure sufficient source code to understand how the accused devices and instrumentalities operate. The cycle is as follows: SEVEN asks Google to produce *complete* source code for the components at issue. Google then requires SEVEN to identify the code it seeks (that SEVEN has never seen) with exacting detail. Google produces only *some* of the source code, removing files and directories it contends are irrelevant. SEVEN then sends its experts to review the partial productions, where they discover missing files.[1] In turn, SEVEN again asks Google to produce *complete* source code for the relevant components. Google again demands that SEVEN request specific code, and again

[1] For example, during SEVEN's most recent review, it discovered production deficiencies with regard to the device-side code and server-side Google Play code. Specifically, with regard to the device-side code, Google did not produce Play Movies & TV, Play Music, Play Books, or their respective players. And with regard to the server-side Google Play code, Google failed to produce code for [redacted] related to providing app downloads; [redacted] related to providing app downloads (including the code relating to providing and/or handling [redacted] server code related to providing app downloads (e.g., in the 2011 version of Google Play); or backend server code for delivering content through Play Movies & TV.

[redacted]

provides piecemeal productions from which SEVEN cannot develop an understanding of the functionalities. SEVEN has repeated this cycle close to half a dozen times. With the close of discovery rapidly approaching, SEVEN can no longer afford to repeat this pattern.

ARGUMENTS AND AUTHORITIES

A. Legal standard: An accused infringer must produce all source code reasonably needed for a plaintiff to understand *for itself* how the technology at issue operates and functions.

The Federal Rules of Civil Procedure provide for broad discovery—parties are entitled to discover any non-privileged matter relevant to any party's claim or defense in proportion to the needs of the case. *See* Fed. R. Civ. P. 26(b)(1). Further, this District's Patent Rules mandate the production of source code. Specifically, a party opposing a patent-infringement claim *must* produce "[s]ource code . . . sufficient to show the operation of any aspects or elements of an accused instrumentality identified by the patent claimant in its [infringement contentions]," at the same time it serves its preliminary invalidity contentions. *See* P.R. ¶ 3-4. This rule "imposes an affirmative obligation on the accused infringer to produce source code and all other relevant materials reasonably needed for the Plaintiff to understand *for itself* how the technology at issue operates and functions." *Edward D. Ioli Trust*, 2012 WL 5830711, at *2 (emphasis in original). While the amount of code actually produced "may be larger than what will likely be used at trial . . . the scope of relevant discovery is broader (often much broader) than the evidence that is ultimately admitted at trial." *See MobileMedia Idaes LLC v. HTC Corp.*, No. 2:10–cv–112–JRG, 2012 WL 7783403, at *2 (E.D. Tex. Oct. 19, 2012).

The burden is on "[t]he party resisting discovery"—here, Google—"[to] show specifically how each discovery request is not relevant or [is] otherwise objectionable." *See McKinney/Pearl Rest. Partners, L.P. v. Metropolitan Life Ins. Co.*, No. 3:14-cv-2498-B, 2016 WL 2997744, at *4

[redacted]

(N.D. Tex. May 25, 2016) (citing *McLeod, Alexander, Powel & Apffel, P.C. v. Quarles*, 894 F.2d 1482, 1485 (5th Cir. 1990)). When a party fails to produce source code, the party seeking the discovery may move to compel a production or inspection. *See* Fed. R. Civ. P. 37(a)(3); *see also Edward D. Ioli Trust*, 2012 WL 5830711, at *3 (compelling source code production). "An evasive or incomplete disclosure, answer, or response must be treated as a failure to disclose, answer, or respond." Fed. R. Civ. P. 37(a)(4). Here, Google cannot show that SEVEN's requests for source code are irrelevant or otherwise objectionable.

B. Google must produce all source code necessary for SEVEN to understand how the accused devices and instrumentalities operate.

Google's only objection to providing all the code SEVEN seeks is that SEVEN allegedly seeks irrelevant and disproportional information. It does not. To the contrary, SEVEN no more wants to sift through millions of lines of irrelevant source code than Google wants to produce it. Accordingly, SEVEN has tailored its requests to seek only the code reasonably necessary to show how the accused devices and instrumentalities operate.

1. SEVEN seeks relevant code necessary to its case.

SEVEN seeks only relevant code "reasonably needed for [SEVEN] to understand *for itself* how the technology at issue operates and functions." *See Edward D. Ioli Trust*, 2012 WL 5830711, at *2; *see also* E.D. Tex. L.R. CV-26(d)(4)(defining relevant information as, among other things, "information that deserves to be considered in the preparation, evaluation or trial of a claim or defense"). Specifically, SEVEN seeks complete copies of all versions[2] of device-side

[2] The only versions for which Google has produced code are the versions at the beginning of the infringement period, and the latest version. But SEVEN has accused all versions in between those first and last versions. It is thus entitled to the code for each. *See Apple Inc. v. Samsung Elecs. Co.*, No. C 11–1846 LHK (PSG), 2012 WL 1595784, at *2 (N.D. Cal. May 4, 2012) (order

[redacted]

code for accused Nexus and Pixel products; application code for Google Drive, Google Maps, and Google+; and server code for Gmail, the Google Play Store, and the Google Play Console. As SEVEN has explained to Google, this code is directly relevant to SEVEN's understanding of the accused devices and functionalities:

Requested Code	Relevance
Device-side code for Nexus and Pixel products	SEVEN has explicitly accused Nexus and Pixel products of infringing seven of the ten asserted patents. The entire code on the devices is therefore necessary for SEVEN to develop a complete understanding of how the products operate. Notably, in the consolidated case, **Samsung has produced all requested device-side code installed on its products. ZTE has committed to doing the same in the Northern District case.**
Application code for Google Drive, Google Maps, and Google+	Relevant to infringement of at least the '952, '812, '127, and '600 patents. Specifically, these apps use the accused functionalities, [redacted] In fact, SEVEN's Infringement Contentions explicitly identify Google Drive, Google Maps, and Google+ as products that use the accused 2-Step Verification functionality.[3]
Gmail server code	Relevant to infringement of at least the '952 patent. For example, claim 26 requires that the accused products "exchange transactions with a client operating in a network through a connection provided *through a server* coupled to the client." SEVEN's Infringement Contentions explain how the accused products exchange transactions relating to the Gmail application with clients, such as Google software running on Google's Gmail servers.
Google Play Store server code	Relevant to infringement of at least the '433 and '158 patents. In fact, SEVEN has explicitly accused "[a]ll versions of the Google Play Store, *including the Google Play Store servers*" of infringing the '433 patent. The complete source code for the Play Store server is relevant to tracing which portions of the

compelling source-code production included "all relevant Samsung source code versions, and not only the release-version source code that Samsung deemed most relevant.").

[3] *See, e.g.*, **Ex. A**: SEVEN's Infringement Contentions, Exhibit J at 1.

[redacted]

	produced code are called and used; without it, SEVEN cannot determine how the Play Store is configured to operate.
Google Play Console server code	Relevant to infringement of at least the '158 patent. For example, claim 10 of the '158 patent requires that certain information be provided to a network server for use in determining requirements for operating an application on a mobile device. SEVEN's Infringement Contentions explain that Google Play servers filter apps based on several types of information. Complete source code for the Google Play Console server is relevant to the [redacted] in the Google Play Console.

Rather than provide the entire code for these devices and functionalities, Google has provided select files and directories, withholding other files required for the components at issue to function.[4] But the entire code is critical as it "not only provides the blueprint for how the software components work, but also often contains developer comments that describe the intent and purpose of various sub components of the software."[5] As SEVEN's expert, Dr. Michael Goodrich, explains: Google's "[p]artial source code productions interfere with my analysis, because they prevent a complete understanding of how the components in a system function. . . . [and] getting missing files through this piecemeal process can needlessly lead to delays, multiple review sessions, and incomplete analyses."[6] Accordingly, "the simplest, fastest, and least error-prone method is to provide the complete source code for the requested components."[7]

SEVEN does not seek "to review all code of any kind in [Google's] possession, just that code which will allow [SEVEN] to determine, for itself, the functionality and operation of the . . .

[4] *See* **Ex. B**: Declaration of Michael Goodrich ¶ 3.

[5] *See* **Ex. B**: Declaration of Michael Goodrich ¶ 2.

[6] *See* **Ex. B**: Declaration of Michael Goodrich ¶ 4.

[7] *See* **Ex. B**: Declaration of Michael Goodrich ¶ 5.

[redacted]

accused products outlined in [this] [M]otion." *See, e.g.*, *Adaptix, Inc. v. Alcatel-Lucent USA, Inc.*, No. 6:12-CV-22, 2014 WL 11538105, at *2–3 (E.D. Tex. June 2, 2014) (compelling production of "all source code related to" certain identified features); *see also Apple*, 2012 WL 1595784, at *2 (awarding sanctions for failure to "produce all source code for all accused products"). Certainly, the total amount of code produced "may be larger than what will likely be used at trial." *See MobileMedia Idaes LLC*, 2012 WL 7783403, at *2. But it is nevertheless relevant and discoverable. *See id.* SEVEN therefore asks the Court to compel its production.

2. Google has created burden by making misplaced relevance objections.

Google has also suggested that SEVEN seeks an amount of code disproportional to this case's needs. But SEVEN's requests are proportional, especially in view of the number of patents and accused instrumentalities and devices in this case. *Cf. MobileMedia Ideas LLC*, 2012 WL 7783403, at *2 ("Given the enormous size of the source code reviewed by HTC, the Court does not find MMI's 7,700 page print request unreasonable."). Further, any alleged burden is of Google's own making. In fact, as Google's counsel admitted on meet-and-confers, parsing and extracting which bits and pieces of code to produce to SEVEN requires considerable effort—that process is significantly more difficult than it would be simply for Google to produce all of the code.

CONCLUSION

> In a typical patent infringement case . . . few tasks excite a defendant less than a requirement that it produce source code. . . . [But] there is no source code exception to the production requirements of [Rule] 34. And so, subject to the proportionality and burden considerations . . . , when a patentee requests source code for one or more accused products, a defendant must produce it.

Apple, 2012 WL 1595784, at *1. Because no source code exception applies to the code SEVEN seeks from Google, SEVEN asks the Court to compel Google to produce it.

[redacted]

Dated: August 1, 2018

Respectfully submitted,

/s/ Justin S. Cohen

Bruce S. Sostek
State Bar No. 18855700
Bruce.Sostek@tklaw.com
Max Ciccarelli
State Bar No. 00787242
Max.Ciccarelli@tklaw.com
Herbert J. Hammond
State Bar No. 08858500
Herbert.Hammond@tklaw.com
Richard L. Wynne Jr.
State Bar No. 24003214
Richard.Wynne@tklaw.com
J. Michael Heinlen
State Bar No. 24032287
Michael.Heinlen@tklaw.com
Adrienne E. Dominguez
State Bar No. 00793630
Adrienne.Dominguez@tklaw.com
Justin Cohen
State Bar No. 24078356
Justin.Cohen@tklaw.com
Vishal Patel
State Bar No. 24065885
Vishal.Patel@tklaw.com
Nadia E. Haghighatian
State Bar No. 24087652
Nadia.Haghighatian@tklaw.com
Austin Teng
State Bar No. 24093247
Austin.Teng@tklaw.com
Natalie M. Cooley
State Bar No. 24079912
Natalie.Cooley@tklaw.com
Matthew Cornelia
State Bar No. 24097534
Matt.Cornelia@tklaw.com

THOMPSON & KNIGHT LLP
One Arts Plaza
1722 Routh St., Suite 1500
Dallas, Texas 75201
214.969.1700
214.969.1751 (Fax)

Samuel F. Baxter
Texas State Bar No. 01938000
sbaxter@mckoolsmith.com

MCKOOL SMITH, P.C.
104 E. Houston Street, Suite 300
Marshall, Texas 75670
Telephone: (903) 923-9000
Facsimile: (903) 923-9099

Theodore Stevenson, III
Texas State Bar No. 19196650
tstevenson@mckoolsmith.com
Eric S. Hansen
Texas State Bar No. 24062763
ehansen@mckoolsmith.com

MCKOOL SMITH, P.C.
300 Crescent Court, Suite 1500
Dallas, Texas 75201
Telephone: (214) 978-4000
Telecopier: (214) 978-4044

ATTORNEYS FOR PLAINTIFF
SEVEN NETWORKS LLC

[redacted]

CERTIFICATE OF SERVICE

I certify that on August 1, 2018, a true and correct copy of the foregoing document was served on all counsel of record via ECF.

/s/ Justin Cohen
Justin Cohen

CERTIFICATE OF CONFERENCE

I certify that on July 24, 2018, I and Sam Baxter (SEVEN) conferred via telephone with Charlie Verhoeven and Mike Jones (Google) in compliance with L.R. CV-7(h). This phone call was the culmination of months of written and telephonic correspondence between the Parties in attempts to resolve their dispute. Despite these efforts, the Parties have reached an impasse regarding the relief sought. Court assistance is therefore necessary. Google opposes this Motion.

/s/ Max Ciccarelli
Max Ciccarelli

/s/ Samuel F. Baxter
Samuel F. Baxter

CERTIFICATE OF AUTHORITY TO FILE UNDER SEAL

I certify that I have authority to file this document under seal under this Court's Local Rules and the Protective Order in this case.

/s/ Justin Cohen
Justin Cohen

Exhibit A

EXHIBIT J: INFRINGEMENT CONTENTIONS FOR U.S. PATENT NO. 9,444,812

Below is claim chart identifying specifically where each element of each asserted claim of U.S. Patent No. 9,444,812 is found within the Google '812 Accused Product (also referred to in this chart as "Accused Product"). All references to specific versions of the Google '812 Accused Product are merely exemplary in nature. Accordingly, neither the claim chart nor the contentions are limited to just those specific examples provided in the chart. Rather, the claim chart is representative of the contentions as to all versions of the Google '812 Accused Product.

Claim Language	Infringement Contentions for Accused Product
1. A method for authenticating a user to provide a service, the method comprising:	The preamble is not limiting, but to the extent that it is determined that the preamble is a limitation of this claim, Google's 2-Step Verification process is a method for authenticating a user to provide a service. Google provides to its users a number of services associated with Google's products, such as Gmail, Google Play, Google Maps, Google+, and Google Drive. *See, e.g.*, *Our Products*, Google, https://www.google.com/intl/en/about/products (last visited July 1, 2017).

Exhibit B

UNITED STATES DISTRICT COURT
FOR THE EASTERN DISTRICT OF TEXAS
MARSHALL DIVISION

SEVEN NETWORKS, LLC, Plaintiff, v. GOOGLE LLC, Defendant.	CIVIL ACTION NO. 2:17-CV-442-JRG LEAD CASE PATENT CASE JURY TRIAL DEMANDED
v. SAMSUNG ELECTRONICS AMERICA, INC. AND SAMSUNG ELECTRONICS CO., LTD., Defendants.	CIVIL ACTION NO. 2:17-CV-441-JRG CONSOLIDATED CASE

DECLARATION OF MICHAEL T. GOODRICH, PHD
REGARDING INCOMPLETE GOOGLE SOURCE CODE

I, Michael T. Goodrich, declare under penalty of perjury as follows:

1. I am currently a Chancellor's Professor in the Department of Computer Science at University of California, Irvine. I have been retained by SEVEN Networks, LLC to investigate whether Google mobile devices infringe certain patents-in-suit that concern mobile device software. Google mobile devices include multiple relevant software components, such as Google Cloud Messaging, Firebase Cloud Messaging, Gmail, and the Google Play Store. In addition, several of SEVEN's patents-in-suit concern two-factor authentication and mobile application provisioning.

2. Reviewing the source code for these software components aids my analysis. Source code not only provides the blueprint for how the software components work, but also often contains developer comments that describe the intent and purpose of various sub components of the software.

3. On June 28, 2018, I inspected Google source code at Quinn Emanuel's San Francisco office to determine whether the complete source for the request components had been provided. I observed the following for at least Firebase Cloud Messaging (FCM) and 2FA (two-factor authentication): [redacted]

4. Partial source code productions interfere with my analysis, because they prevent a complete understanding of how the components in a system function. For example, a common, repeated source code review exercise involves tracing logical execution flows through a software component from a known entry point. This usually involves tracing method calls in one file to the file(s) where those methods are defined, and then recursively applying this same approach in succession down through the chain of method calls until reaching a low enough conceptual level to have sufficient confidence in how the source code functions. Missing source code files hamper this analysis by cutting short my ability to follow such call chains. Furthermore, determining and requesting specific missing files and then receiving the requested files may only resolve one step in a call chain; hence, getting missing files through this piecemeal process can needlessly lead to delays, multiple review sessions, and incomplete analyses.

5. In my view, it is unlikely that Google would be able to identify all of the files that I would consider to be relevant to my analysis. Source code files are typically highly interdependent and require judgment to separate relevant code from irrelevant code on a file-by-file basis. In judging relevance, I may reach different conclusions than those of Google's employees or its experts. Therefore, the simplest, fastest, and least error-prone method is to provide the complete source code for the requested components.

Dated: July 30, 2018

Michael T. Goodrich, PhD